REMARKS/ARGUMENTS

Claims 1-39 are pending in the present application. No claims were canceled or added. Claims 1, 2, 3, 8, 14, 15, 21, 27-29, and 34 were amended. Amendments to claims 3, 21, and 29 were made in response to antecedent basis and statutory rejections and not in response to any art rejection. These amendments have not changed in scope with respect to whether the claims are patentable over the cited references. Support for the amendments to the claims can be found in the Specification at least on page 7, page 15-16, page 19, page 21, and page 28. Reconsideration of the claims is respectfully requested.

I. 35 U.S.C. § 112, First Paragraph: Claims 1-39

The examiner has objected to the specification under 35 U.S.C. § 112, first paragraph, as failing to adequately teach how to make and/or use the invention in claims 1-39. Additionally, the examiner rejected the claims under the same reasons. This rejection is respectfully traversed. Independent claims 1, 14, and 27 are amended to more clearly reflects the wording used in the disclosure. In addition, each and every limitation of independent claims 1, 14, and 27 are supported in the specification. For example, specific references to the Specification supporting each claim limitation in independent claim 1 are provided as follows:

> 1. A computer implemented method in a data processing system for automatically configuring IP security tunnels (page 19, lines 22-26; page 1, lines 4-6), said computer implemented method comprising the steps of:
>
> retrieving a remote computer system identifier (page 15, lines 25-31);
>
> determining whether a local-remote pair corresponding to the identifier is found (page 15, line 29- page 1; page 16, lines 4-6), wherein the local-remote pair is used in selecting a security policy (page 16, lines 2-4), and wherein an error is reported indicating that a user needs to configure a tunnel with the remote computer system if the local-remote pair is not found (page 16, lines 6-14); and
>
> defining a configuration of an IP security tunnel between the data processing system and the remote computer system utilizing a security policy specification format (page 1, lines 8-14; page 13, lines 11-12), wherein said security policy specification format is established as a document type definition (DTD) file (page 19, lines 26-31; page 28, lines 7-11; page 33, lines 14-16) capable of being utilized by a plurality of different operating systems and a plurality of different machine types (page 1, lines 8-14; page 19, lines 22-26; page 33, lines 7-10).

The citations to the specification supporting the claim language are given only as examples of where the claim language in claim 1 is supported and not meant to limit the claims to the particular examples using the claim language in the specification. Therefore, the objection of the specification under 35 U.S.C. § 112, first paragraph as failing to adequately teach how to make and/or use the invention in claims 1-39 has been overcome.

II. 35 U.S.C. § 112, Second Paragraph: Claims 2-4, 15, and 28-30

The examiner has rejected claims 2-4, 15, and 28-30 under 35 U.S.C. § 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter, which applicants regard as the invention. These claims have been amended to overcome the rejections. Therefore, the rejection of claims 2-4, 15, and 28-30 under 35 U.S.C. §112, second paragraph have been overcome.

III. 35 U.S.C. § 102, Anticipation: Claims 1, 5-14, 18-27, and 31-39

The examiner has rejected claims 1, 5-14, 18-27, and 31-39 under 35 U.S.C. § 102(e) as being anticipated by *D'Sa* et al., System and Method for Multiple Virtual Private Network Authentication Schemes, U.S. Patent Publication No. 2002/0178355, November 28, 2002 (hereinafter "*D'Sa*"). This rejection is respectfully traversed.

Independent Claims 1, 14, and 27

The rejection states:

> As per claims 1,14, and 27, the applicant describes a data processing system for defining a configuration of IP security tunnels comprising the following limitations which are met by *D'Sa*:
> a) exchanging identification data with a remote computer system ([0041],[0047]-[0048],Fig 2);
> b) determining, based on the identification data, whether a predefined security policy exists corresponding to the remote computer system (([0041],[0047]-[0048],Fig 2);
> c) selecting a security policy specification format capable of being utilized by a plurality of different operating systems and a plurality of different machine types if a predefined security policy is absent ([0041],[0047]-[0048],Fig 2);
> d) said system for automatically configuring an IP security tunnel between the data processing system and the remote computer system utilizing said security policy specification format ([0042],[0047]-[0048],Fig 2).

Office Action dated January 31, 2006, pps. 4-5.

A prior art reference anticipates the claimed invention under 35 U.S.C. § 102 only if every element of a claimed invention is identically shown in that single reference, arranged as they are in the claims. *In re Bond*, 910 F.2d 831, 832, 15 U.S.P.Q.2d 1566, 1567 (Fed. Cir. 1990). All limitations of the claimed invention must be considered when determining patentability. *In re Lowry*, 32 F.3d 1579, 1582, 32 U.S.P.Q.2d 1031, 1034 (Fed. Cir. 1994). Anticipation focuses on whether a claim reads on the product or process a prior art reference discloses, not on what the reference broadly teaches. *Kalman v. Kimberly-Clark Corp.*, 713 F.2d 760, 218 U.S.P.Q. 781 (Fed. Cir. 1983). In this particular case, the cited reference

does not teach or disclose each and every feature of the presently claimed invention as they are arrange in the claims.

Amended claim 1 reads as follow:

> 1. A computer implemented method in a data processing system for automatically configuring IP security tunnels, said computer implemented method comprising the steps of:
>
> retrieving a remote computer system identifier;
>
> determining whether a local-remote pair corresponding to the identifier is found, wherein the local-remote pair is used in selecting a security policy, and wherein an error is reported indicating that a user needs to configure a tunnel with the remote computer system if the local-remote pair is not found; and
>
> defining a configuration of an IP security tunnel between the data processing system and the remote computer system utilizing [[said]] a security policy specification format, wherein said security policy specification format is established as a document type definition (DTD) file capable of being utilized by a plurality of different operating systems and a plurality of different machine types.

In particular, *D'Sa* does not teach the defining step in amended claim 1. More specifically, *D'Sa* does not teach "defining a configuration of an IP security tunnel between the data processing system and the remote computer system utilizing a security policy specification format, wherein said security policy specification format is established as a document type definition (DTD) file capable of being utilized by a plurality of different operating systems and a plurality of different machine types," as is recited in claim 1. The Examiner has cited to *D'Sa* at paragraph [0041], [0042], [0047], [0048], and Figure 2.

Figure 2 of *D'Sa* is shown as follows:

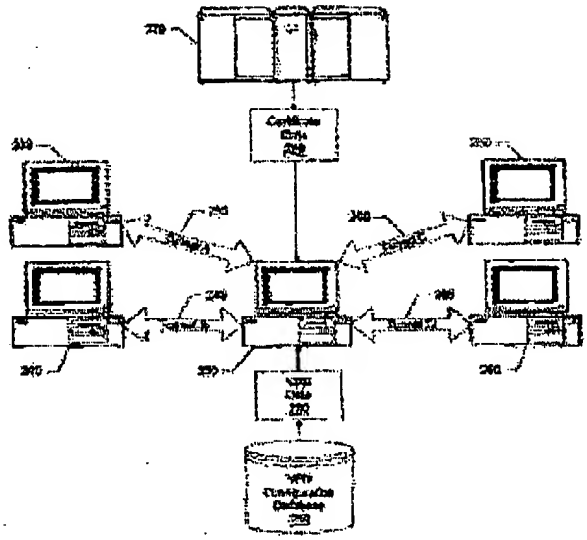


Figure 2

As can be seen, **Figure 2** merely illustrates multiple existing tunnels connecting multiple computers. **Figure 2** does not show defining a security tunnel configuration or utilizing a security policy specification format for defining the tunnel configuration. Moreover, a security policy of any kind is not shown in the Figure. Finally, the figure does not disclose a security policy format that is a document type definition file, as is claimed in claim 1.

The examiner also points to the following portion of *D'Sa* :

> [0041] In the example shown, computer system 200 establishes tunnel A 235 securely connecting first computer system 230 with computer system 200. Likewise, tunnel B 245 securely connects second computer system 240 with computer system 200, tunnel C 255 securely connects third computer system 250 with computer system 200, and tunnel D 265 securely connects fourth computer system 260 with computer system 200. Each of these computer systems, 230, 240, 250, and 260, have identification information and authentication information stored in VPN configuration database 210.

D'Sa, paragraph [0041].

Here, *D'Sa* states that Figure 2 shown above shows a computer establishes a tunnel with a first computer, a second computer, a third computer, and a fourth computer. Each computer has identification information and authentication information stored in a virtual private network (VPN) configuration database. Although *D'Sa* may disclose a configuration database for storing identification information for multiple different computers, *D'Sa* does not teach a security policy specification format that is a DTD file capable of being utilized by a plurality of different operating systems and machine types. Furthermore,

this section of *D'Sa* does not even mention defining a configuration of a security tunnel using a security policy format of any kind, let alone a format that is a document type definition file.

The Examiner also cites to *D'Sa* at paragraph [0042] which states:

> [0042] FIG. 3 shows a database diagram of tables used in configuring tunnels between the computer and other computer systems. VPN configuration database 300 is shown with four tables. Endpoints table 310 includes a list of configured tunnels between the computer system and other computer systems. One end of each endpoint identifies the computer system, while the other end of the endpoint identifies a remote computer. Each of the computers included in endpoints table 310 is identified with an identifier, such as an address. In addition, endpoints table 310 includes IP addresses for the remote computer systems. An IP address is an identifier for a computer or device on a TCP/IP network. Networks using the TCP/IP protocol route messages based on the IP address of the destination. The format of an IP address is a 32-bit numeric address written as four numbers separated by periods. Each number can be zero to 255. For example, 1.160.10.240 could be an IP address. Within an isolated network, IP addresses can be assigned at random so long as each one is unique. However, connecting a private network to the Internet requires using registered IP addresses (called Internet addresses) to avoid duplicates. The four numbers in an IP address are used in different ways to identify a particular network and a host on that network. Finally, endpoints table 310 includes a flag indicating whether a Certificate Revocation List (CRL) is used to check whether a given certificate has been revoked. Other valid ID types include FQDN, user@FQDN, distinguished names, and key IDs.

D'Sa, paragraph [0042].

As can be seen, this cited portion of *D'Sa* discloses a configuration database containing a list of configured tunnels between a computer system and a remote computer system to determine a compatible authentication system. Although *D'Sa* appears to teach use of information in a configuration database for selecting an access method for computer systems requesting a VPN, this section of *D'Sa* does not teach a security policy specification format that is a DTD file capable of being utilized by a plurality of different operating systems and a plurality of different machine types. In fact, a security policy format and/or a DTD file is not even mentioned in this or any other section of the reference. Moreover, this section of the reference merely states that a list of configured tunnels is contained in a configuration database. Such statements do not disclose **defining** a configuration of an IP security tunnel utilizing a security policy **specification format.**

D'Sa also states:

> [0047] FIG. 4 shows a flowchart of the creation of a tunnel using VPN configuration data. Processing commences at 400 whereupon a remote computer identifier is retrieved (input 405) corresponding to a remote computer to be connected in a VPN with the current computer system. The remote computer ID is typically received from a user command or IKE message. The remote computer ID is retrieved for both the initiator and the responder. The local-remote endpoints pair corresponding to the remote computer system identifier and the local computer identifier is selected from the endpoints table (step 410). The ID Rules List links the local-remote endpoints pair to a security policy name that is used in selecting the security policy (see step 440). A determination is made

> as to whether the endpoints pair was found (decision 415). If the pair was not found, decision 415 branches to "no" branch 420 whereupon an error is reported that the user needs to configure a tunnel with the remote computer system before the tunnel can be used (step 425) and processing terminates (end 430). Additionally, step 425 could invoke a configuration screen allowing the user to configure the tunnel with the remote computer by supplying the needed access information.

D'Sa, paragraph [0047].

Here, *D'Sa* states that a remote computer identifier is retrieved. An endpoints pair corresponding to the identifier is selected. If the endpoints pair is not found, an error is reported. This section of *D'Sa* does not teach **defining** a configuration of an IP security tunnel utilizing a security policy specification format that is a DTD file capable of being utilized by a plurality of different operating systems and a plurality of different machine types, as is claimed in claim 1. In fact, a DTD file is not even mention in this or any other section of the reference.

The Examiner also cites to *D'Sa* at paragraph [0048], which states:

> [0048] If the pair was found in the endpoints table, decision 415 branches to "yes" branch 435 whereupon a policy corresponding to the local-remote pair is selected from the policy table (step 440). The policy includes a proposal list with separate initiator and responder proposals. Proposals have general characteristics, like lifetimes and transform names. Transforms include specific encryption algorithms, hash algorithms, and authentication methods being proposed. A determination is made as to whether a corresponding policy was found (decision 445). If a corresponding policy was not found, decision 445 branches to "no" branch 450 whereupon a default policy is used (step 455). For example, a default policy could be used to use a digital certificate (if available), before attempting to use any available pre-shared keys. If the policy is found, decision 445 branches to "yes" branch 460.

D'Sa, paragraph [0048].

As shown above, *D'Sa* merely teaches selecting a security policy for a particular local computer and remote computer pair based on an endpoint pair. *D'Sa* does not teach a security policy **specification format** capable of being utilized by a plurality of different operating systems and a plurality of different machine types, as opposed to a single pair of computers. In addition, this section of the reference does not mention defining a configuration of a security tunnel or a security policy specification format utilized to define a security tunnel configuration that is a document type definition file. Therefore, *D'Sa* fails to disclose "defining a configuration of an IP security tunnel between the data processing system and the remote computer system utilizing a security policy specification format, wherein said security policy specification format is established as a document type definition (DTD) file capable of being utilized by a plurality of different operating systems and a plurality of different machine types," as is recited in claim 1.

Therefore, *D'Sa* fails to teach each and every feature recited in amended claim 1. Independent claims 14 and 27 include similar features and are distinguishable over *D'Sa* based on the same rationale set forth above with regard to claim 1.

Dependent Claims 5-13, 18-26, and 31-39

By virtue of their dependency on independent claims 1, 14, and 27, *D'Sa* does not teach each and every feature of dependent claims 5-13, 18-26, and 31 -39. Additionally, claims 5-13, 18-26, and 31-39 claim other additional combinations of features not suggested by the reference.

For example, with respect to claims 6, 11, 16, 24, 29, and 37, the Examiner alleges that *D'Sa* discloses a protection element at paragraph [0099], which states as follows:

> Depending on the authentication method used, key values are fetched from Public/Private Keys database 740 and Pre-Shared Keys database 745. For authentication methods that use public key encryption, Public/Private Keys database 740 is used. The Public/Private Keys database includes local private keys and corresponding digital certificates which contain the corresponding public key of the local ID and signing certificates including public keys corresponding to the signing certificates.

D'Sa, [0099].

This section of *D'Sa* discloses fetching key values from a Public/Private Keys database and a Pre-Shared Keys database for authentication. However, there is nothing in this, or any other section of *D'Sa*, that teaches "a protection element in said security policy specification format, said protection element including a listing of IKE transforms," as is recited in claims 6, 11, 16, 24, 29, and 37.

As to claims 11, 24, and 37 the Examiner alleges that *D'Sa* teaches an IPsec proposal element, an IPsec authentication header element, and an IPsec protection element at paragraphs [0071], [0072], and [0146], which are as follows:

> Initiator Proposal List Index—an index to a initiator proposal list record (see Proposal List 725, below). If the Initiator Proposal List Index is null then initiation with the remote ID is not allowed (i.e., the system only acts as a responder to the remote ID).
>
> Responder Proposal List Index—an index to a responder proposal list record (see Proposal List 725, below). If this value is null, then response is not allowed (i.e., system only acts as an initiator when dealing with the remote ID). If both the Initiator Proposal List Index and the Responder Proposal List Index values are null, then no negotiation is allowed between the systems.

D'Sa, [0071]-[0072].

> The number authentication header (AH) Transforms, if this value is 0 then AH will not be proposed.

D'Sa, [0146].

Here, *D'Sa* discloses an Initiator Proposal List Index and a Responder Proposal List Index regarding the method of *D'Sa* by which an initiating computer proposes one or more authentication

methods and a responder computer selects an authentication method from the initiator's proposal list. The above cited portions of *D'Sa* does not disclose any teachings regarding the Internet Protocol Security Protocol (IPsec) or "an IPsec proposal element, an IPsec ESP protocol element, an IPsec authentication header element, and an IPsec protection element" in a security policy specification format, as is claimed in claims 11, 24 and 37.

In regards to claims 12-13, 25-26, and 38-39, the Examiner states that *D'Sa* describes the step of automatically configuring an IP security tunnel utilizing the security policy specification format at *D'Sa*, paragraphs [0040] and [0041], which are set forth above. Here, *D'Sa* discloses establishing a VPN between a computer system and remote computer systems by using a configuration database containing identification data and authentication information for the computer system and the remote computer systems. However, *D'Sa* does not disclose a standardized format or "automatically configuring an IP security tunnel utilizing said security policy specification format," as is recited in claims 12, 15, and 38.

Regarding claims 8, 21, and 34, *D'Sa* does not teach "establishing a group element in said security policy specification format, wherein said group element contains multiple identification elements." The Examiner believes this feature is discloses by *D'Sa* at paragraph [0065] which states as follows:

Group Name-a unique logical name that can be used as a database search key.

D'Sa, paragraph [0065].

This section of *D'Sa* merely mentions a group name that may be included in a group. *D'Sa* does not teach a group element in a security policy specification format that contains multiple identification elements, as is claimed in dependent claims 8, 21, and 34.

Consequently, it is respectfully urged the rejection of claims 1, 5-14, 18-27 and 31-39 under 35 U.S.C. § 102 has been overcome.

Furthermore, *D'Sa* does not teach, suggest, or give any incentive to make the needed changes to reach the presently claimed invention. *D'Sa* actually teaches away from the presently claimed invention because it teaches utilization of a list of already configured tunnels in the Endpoints table of the configuration database and preference data for use in determining a compatible access policy as opposed to defining a configuration of an IP security tunnel utilizing a security policy specification format that is a document type definition file capable of being utilized by a plurality of different operating systems, as in the presently claimed invention. Absent the Examiner pointing out some teaching or incentive to implement *D'Sa* and utilizing a standard security policy specification format that is a DTD file to configure IP security tunnels, one of ordinary skill in the art would not be led to modify *D'Sa* to reach the present invention when the reference is examined as a whole. Absent some teaching, suggestion, or incentive to modify *D'Sa* in this manner, the presently claimed invention can be reached only through an

improper use of hindsight using the applicants' disclosure as a template to make the necessary changes to reach the claimed invention.

IV. 35 U.S.C. § 102, Anticipation: Claims 1, 14, and 27

The examiner has rejected claims 1, 14, and 27 under 35 U.S.C. § 102(e) as being anticipated by *Bendinelli* et al., Methods and Systems for Enabling a Tunnel Between Two Computers on a Network, U.S. Patent No. 6,631,416, October 7, 2003 (hereinafter "*Bendinelli*"). This rejection is respectfully traversed.

The rejection states:

> As per claims 1,14, and 27, the applicant describes a data processing system for defining a configuration of IP security tunnels with the following limitations which are met by *Bendinelli*:
>
> a) exchanging identification data with a remote computer system (Col 17, lines 21-65; Col 10, line 60 to Col 11, line 8);
>
> b) determining, based on the identification data, whether a predefined security policy exists corresponding to the remote computer system (Col 17, lines 21-65; Col 10, line 60 to Col 11, line 8);
>
> c) selecting a security policy specification format capable of being utilized by a plurality of different operating systems and a plurality of different machine types if a predefined security policy is absent (Col 17, lines 36-63);
>
> d) said system for automatically configuring an IP security tunnel between the data processing system and the remote computer system utilizing said security policy specification format (Col 17, lines 36-63).

Office Action dated January 31, 2006, page 6.

Bendinelli does not teach "defining a configuration of an IP security tunnel between the data processing system and the remote computer system utilizing a security policy specification format, wherein said security policy specification format is established as a document type definition (DTD) file capable of being utilized by a plurality of different operating systems and a plurality of different machine types," as is recited in claim 1.

The Examiner cites to *Bendinelli* at column 17, lines 36-63, which states as follows:

> FIG. 3 shows an exemplary flowchart for initially registering one or more gateways with the control system 175. Referring to FIGS. 1 and 3, the user may register at least one of the gateways 150-153 with the control system 175 (step 310) and define a configuration for the registered gateways 150-153 (step 320). In one embodiment, the user may contact the control system 175 through the Internet using a web browser to specify a particular configuration for a gateway. This specified configuration information may include a name for the gateway and a name for the virtual private network. This name for the virtual private network will hereinafter be referred to as the virtual private network's domain name.
>
> The control system 175 may use the specified configuration to assemble code and information, such as program code and textual information (e.g., Extensible Markup

> Language also referred to as "XML"), in the form of a disk image (step 330). This disk image may include all the program code and information needed to configure gateways 150-153 for establishing one or more virtual private networks established over communication channel 120. The disk image may then be provided to the user and installed on a processor, such as a personal computer or a general-purpose computer (step 340). When the processor reboots, it uses the information provided in the disk image to configure itself as a gateway capable of establishing secure tunnels to the control system 175.

Bendinelli, column 17, lines 36-63.

Here, *Bendinelli* discloses that a **user provides information** regarding a desired gateway to a **control system that provides code for establishing a gateway** at the user's computer. Thus, *Bendinelli* teaches a user providing configuration specifications to a control system that defines the configuration in a code provided to the user, rather than a specification format for defining a configuration of a tunnel. *Bendinelli* merely teaches a control system providing code to a user defining a configuration of a gateway. These disclosures are contrary to utilization of **a security policy specification format for defining a configuration of an IP tunnel** as recited in amended claim 1. Further, the presently claimed invention in amended claim 1 recites "**defining a configuration** of an IP security tunnel between the data processing system and the remote computer system **utilizing a security policy specification format.**"

Moreover, this section of *Bendinelli* does not teach a specification format that is a DTD file capable of being utilized by a plurality of different operating systems and a plurality of different machine types. *Bendinelli* does not teach or mention a document type definition (DTD) file in this or any other section of the reference. Thus, *Bendinelli* fails to disclose "defining a configuration of an IP security tunnel between the data processing system and the remote computer system utilizing a security policy specification format, wherein said security policy specification format is established as a document type definition (DTD) file capable of being utilized by a plurality of different operating systems and a plurality of different machine types," as is recited in claim 1. Therefore, the rejection of claims 1, 14, and 27 under 35 U.S.C. § 102(e) has been overcome.

V. 35 U.S.C. § 103, Obviousness: Claims 2-4, 15-17, and 28-30

The examiner has rejected claims 2-4, 15-17, and 28-30 under 35 U.S.C. § 103(a) as being unpatentable over *Bendinelli* in view of *Pfeiffer*, Ralf L., XML Tutorials for Programmers, March 2, 1999, (hereinafter "*Pfeiffer*") . This rejection is respectfully traversed.

The rejection states:

> As per claims 2-4,15-17, and 28-30, the applicant describes the system of claims 1,14, and 27, which are met by *Bendinelli* (see above), with the following limitations which are met by *Bendinelli* in view of *Pfeiffer*:

a) Establishing a security policy specification format capable of being utilized by a plurality of different operating systems and a plurality of different machine types (*Bendinelli*: Col 17, lines 36-63);

b) Establishing said security policy specification format being established as a DTD file (*Bendinelli*: Col 17, lines 36-63; *Pfeiffer*: pages 5-6);

Bendinelli discloses all the limitations of independent claims 1,14, and 27. However, *Bendinelli* discloses that the security policy specification format is established as an XML file, not a DTD file. Pfeiffer discloses that a DTD file cammonly stores policy and rules. Combining *Pfeiffer* with *Bendinelli* would allow the security policy specification format to be stored in a DTD file instead of an XML file. It would have been obvious to one of ordinary skill in the art at the time the invention was filed to incorporate the ideas of Pfeiffer with those of *Bendinelli* because a DTD file is another means to store a security policy, specification format and DTD files typically store policy and rules

Office Action dated January 31, 2006, page 7.

A. The Examiner bears the burden of establishing a *prima facie* case.

The Examiner bears the burden of establishing a *prima facie* case of obviousness based on the prior art when rejecting claims under 35 U.S.C. § 103. *In re Fritch*, 972 F.2d 1260, 23 U.S.P.Q.2d 1780 (Fed. Cir. 1992). In this case, the examiner has failed to establish a *prima facie* case of obviousness because the cited references do not teach or suggest the features of the present invention as believed by the examiner and the references cannot be properly modified or combined to reach the presently claimed invention for the reasons stated below.

B. All claim limitations must be considered, especially when missing from the prior art.

In comparing *Bendinelli* and *Pfeiffer* to the claimed invention in claims 2-4, 15-17, and 28-30, the claim limitations of the presently claimed invention may not be ignored in an obviousness determination. Claims 2-4, 15-17, and 28-30 are dependent on independent claims 1, 14, and 27. As shown above, each and every feature of independent claims 1, 14, and 27 are not shown in *Bendinelli*. In particular, *Bendinelli* merely teaches a control service providing a code for establishing a tunnel at a user's computer to a user based on a user's stated specifications regarding the desired tunnel, rather than a security policy specification format defining a configuration of an IP security tunnel. *Bendinelli* also fails to teach or even mention utilizing a document type definition file. Moreover, *Bendinelli* does not suggest or even mention a security policy specification format for defining a configuration of an IP security tunnel, a document type definition file that is capable of utilizing by a plurality of different operating systems and a plurality of different machine types, or the desirability of utilizing such a specification format for defining a security tunnel. Thus, at least by virtue of their dependency on claims 1, 14, and 27, dependent claims 2-4, 15-17, and 28-30 are distinguishable over *Bendinelli* for the same reasons set forth above with regard to independent claims 1, 14, and 27.

Dependent claims 2-4, 15-17, and 28-30 recite additional combinations of features not taught, disclosed or suggested by the cited references. For example, dependent claim 3 recites as follows:

> 3. The method according to claim 1, further comprising:
> including a plurality of different elements in said DTD file, each of said plurality of different elements being utilized to configure an IP security tunnel.

As discussed above, *Bendinelli* does not teach or suggest a security policy specification format capable of being utilizing by a plurality of different operating systems and a plurality of different machine types. Nor does *Bendinelli* teach or suggest including a plurality of different elements in the DTD file to configure an IP security tunnel. In fact, *Bendinelli* teaches away from the presently claimed invention where *Bendinelli* teaches **a control system** utilizing configuration information from a user to **assemble a program code for configuring** a gateway, which is then provided to the user for installation by the user, rather than **a security policy specification format for defining a configuration** of an IP security tunnel.

Moreover, the Examiner admits that *Bendinelli* does not disclose the use of a DTD file, but states that *Bendinelli* discloses a security policy specification format is established as an XML file at column 17, lines 36-63, which is quoted above. As discussed above, *Bendinelli* merely discloses the process whereby a user initially registers with a control system that provides code to the user for establishing a gateway at the user's computer. *Bendinelli* does not provide any teachings, suggestions, or motivations for a security policy specification format for defining a configuration for a gateway. Moreover, *Bendinelli* does not provide any teachings or details as to how the control system configures or defines a configuration for a security tunnel. A security tunnel could be defined by the control system in any number of ways. Furthermore, *Bendinelli* does not teach a user defining a configuration of a security tunnel utilizing a security policy specification format. *Bendinelli* merely discloses a user manually providing information to the control system regarding the desired gateway, such as a name for the gateway. The actual code for establishing the virtual private network is provided to the user by the control system.

Although *Bendinelli* describes a control system providing a program code to establish a VPN that may be provided in XML, such teachings do not amount to a disclosure, suggestion, or motivation for defining a configuration of a security tunnel utilizing a security policy specification format or the security policy format as a DTD file. Moreover, *Bendinelli* does not teach or suggest that a DTD file defines a collection of elements, and generating an XML file utilizing the collection of elements defined in said DTD file, wherein said XML file defines a configuration of a particular IP security tunnel, and wherein said XML file is processed to automatically configure said IP security tunnel defined by the XML file, as is claimed in claims 2, 15, and 28. In fact, the cited portion of the reference does not even mention a DTD file, generating an XML file utilizing the DTD file, a specification format for a security policy that is a DTD file, or any other type of specification format.

Moreover, *Bendinelli's* teachings regarding a control system providing an XML file having code to establish a VPN for a user to execute on a user's computer hardly amounts to a teaching or suggestion to utilize a security policy specification format that is established as a DTD file for defining a configuration of a tunnel or a DTD file for generating an XML file that is processed to automatically configure an IP security tunnel. Therefore, *Bendinelli* fails to teach or suggest the DTD file and XML file claimed in claims 2-4, 15-17, and 28-30.

Pfeiffer fails to make up for the deficiencies of *Bendinelli*. The Examiner recognizes that *Bendinelli* does not disclose a DTD file but believes *Pfeiffer* discloses a DTD file commonly stores policy and rules. However, even if *Pfeiffer* does teach that a DTD file can be used to store policy and rules, such teaching are insufficient to teach or suggest a DTD file as is claimed in claims 2-4, 15-17, and 28-30. The DTD file of *Pfeiffer* does not teach or suggest defining a collection of elements that are utilized to generate an XML file that defines a configuration of a security tunnel. Furthermore, the DTD file of *Pfeiffer* does not teach or suggest defining a configuration of an IP security tunnel utilizing the DTD file or a DTD file capable of being utilized by a plurality of different operating systems and a plurality of different machine types.

In fact, *Pfeiffer* does not provide any teachings, suggestions, or motivations for generating an XML file utilizing a DTD file, defining a configuration of a security tunnel utilizing a DTD file, or a DTD file capable of being utilizing by a plurality of different operating systems and a plurality of different machine types for defining a configuration of an IP security tunnel. Therefore, *Pfeiffer* fails to make up for the deficiencies of *Bendinelli*. Thus, *Bendinelli* and *Pfeiffer*, either alone or in combination, fail to teach or suggest the combination of features recited in claims 2-4, 15-17, and 28-30.

C. The proposed modification would not be made when *Bendinelli* is considered as a whole.

Moreover, the proposed combination of *Bendinelli* and *Pfeiffer* would not be made when *Bendinelli* is considered as a whole. "It is impermissible within the framework of section 103 to pick and choose from any one reference only so much of it as will support a given position, to the exclusion of other parts necessary to the full appreciation of what such reference fairly suggests to one of ordinary skill in the art." *In re Hedges*, 228 U.S.P.Q. 685, 687 (Fed. Cir. 1986). When *Bendinelli* is examined as a whole, *Bendinelli* teaches one of ordinary skill in the art that one who desires to configure a security tunnel registers with a control system that provides code, which may be XML code, to configure the tunnel, rather than utilizing a specification format for defining a configuration for a security tunnel. Therefore, when *Bendinelli* is considered as a whole, the sole purpose taught or suggested for using XML is as a possible form of program code and information provided to user on a disk to establish a VPN.

When *Pfeiffer* is examined as a whole, *Pfeiffer* teaches one of ordinary skill in the art that a DTD file is a grammar that describes what tags and attributes are valid in an XML document. *Pfeiffer* does not

teach or address the problem of establishing a security policy or configuring a security tunnel in a virtual private network. Thus, *Bendinelli* and *Pfeiffer*, when considered as a whole, fail to teach or suggest a security policy specification format as a DTD file for the purpose of defining a configuration of an IP security tunnel. Therefore, the proposed combination of the references would not be made when the references are considered as a whole.

D. Stating that it is obvious to try or make a modification or combination without a suggestion in the prior art is not *prima facie* obviousness.

The Examiner has not provided a proper motivation to combine the different elements from *Pfeiffer i* and *Pfeiffer*. The mere fact that a prior art reference can be readily modified does not make the modification obvious unless the prior art suggested the desirability of the modification. *In re Laskowski*, 871 F.2d 115, 10 U.S.P.Q.2d 1397 (Fed. Cir 1989); see also *In re Fritch*, 972 F.2d 1260, 23 U.S.P.Q.2d 1780 (Fed. Cir. 1992); *In re Mills*, 916 F.2d 680, 16 U.S.P.Q.2d 1430 (Fed. Cir. 1993). The Examiner may not merely state that the modification or combination would have been obvious to one of ordinary skill in the art without pointing out in the prior art a suggestion of the desirability of the proposed modification.

The Examiner states that it would have been obvious to one of ordinary skill in the art at the time the invention was filed to incorporate the ideas of *Pfeiffer* with those of *Bendinelli* because a DTD file is another means to store a security policy specification format and DTD files typically store policy and rules. However, the motivation for the combination of *Bendinelli* and *Pfeiffer* is not based on any rationale, suggestion or motivation provided by the references. There is no teaching or suggestion for establishing a security policy format as a DTD file in any section of either *Bendinelli* or *Pfeiffer*. Thus, the Examiner is merely stating that the references could have been combined without offering any suggestion or motivation provided by the references for the combination of the references.

Even if the references could be properly combined, the combination of the references would not form the presently claimed invention. The present invention is directed towards establishing a security policy specification format as a DTD file for automatically configuring IP security tunnels. A combination of *Bendinelli* and *Pfeiffer* would not form the presently claimed invention in claims 2-4, 15-17, and 28-30. Instead, a combination of the references would merely result in an XML document containing program code and information for a VPN created by a control system in accordance with information provided by a user rather than a security policy specification format that is a DTD file for defining a configuration of an IP security tunnel. Thus, any alleged combination of *Bendinelli* and *Pfeiffer* is not sufficient to form the claimed invention as recited in claims 2-4, 15-17, and 28-30.

E. The claimed invention may only be reached through an improper use of the disclosed invention as a template to piece together and modify the prior art.

Moreover, the Examiner may not use the claimed invention as an "instruction manual" or "template" to piece together the teachings of the prior art so that the invention is rendered obvious. *In re Fritch*, 972 F.2d 1260, 23 U.S.P.Q.2d 1780 (Fed. Cir. 1992). Such reliance is an impermissible use of hindsight with the benefit of applicant's disclosure. *Id.* Therefore, absent some teaching, suggestion, or incentive in the prior art, *Bendinelli* and *Pfeiffer* cannot be properly combined to form the claimed invention. As a result, absent any teaching, suggestion, or incentive from the prior art to make the proposed combination, the presently claimed invention can be reached only through the impermissible use of hindsight with the benefit of applicant's disclosure as a model for the needed changes.

Therefore, the rejection of claims 2-4, 15-17, and 28-30 under 35 U.S.C. § 103(a) has been overcome.

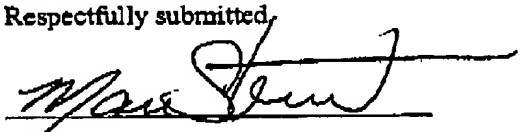
VI. Conclusion

It is respectfully urged that the subject application is patentable over the cited references and is now in condition for allowance.

The examiner is invited to call the undersigned at the below-listed telephone number if in the opinion of the examiner such a telephone conference would expedite or aid the prosecution and examination of this application.

DATE: April 26, 2006

Respectfully submitted,



Mari Stewart
Reg. No. 50,359
Yee & Associates, P.C.
P.O. Box 802333
Dallas, TX 75380
(972) 385-8777
Attorney for Applicants